

OPEN SOURCE RESOURCES FOR TEACHING AND RESEARCH IN MATHEMATICS

Russell Herman and Gabriel Lugo
University of North Carolina Wilmington
hermanr@uncw.edu and lugo@uncw.edu

The high cost of standard mathematical packages is often a hardship for students and faculty (especially in third world countries). As a result, more people are joining the open source movement and are seeking ways to circumvent cost, storage and ownership. We review a diverse number of open source software.

History

While the term “Open Source” has become a popular movement over recent years, it has its origin in the late 1960’s and though the 1970’s. Actually, long before that, software was developed and exchanged freely. In the 1950’s several organizations developed much of the software and it was distributed by computer companies with the hardware. As hardware was the main business, companies sought to keep the price of software low. In 1965 Applied Data Research (ADR) began to license its software at a time when IBM was giving away theirs. However, by the late 60’s ADR filed an antitrust suit against IBM forcing IBM to unbundle most of its software. This led to the commercialization of computer software and operating systems [1].

In 1969 Ken Thompson, Dennis Ritchie, and J.F. Ossanna at AT&T Bell Labs began development of the operating system for the PDP-7 [2]. However, when they were faced with the replacement of the PDP-7 by a PDP-11, they realized that they needed an operating system not tied to the hardware. So, they developed a portable operating system. The operating system was written in the then new “C” language based on the B language of the PDP-7. They decided that operating systems should not depend on the hardware system and that the operating system should be portable between different computers. So, they wrote the new OS and then mailed it out to other users. This led to the birth of a community of Unix hackers. Stallman describes hackers as follows [3]

The use of “hacker” to mean “security breaker” is a confusion on the part of the mass media. We hackers refuse to recognize that meaning, and continue using the word to mean, “Someone who loves to program and enjoys being clever about it.”

Various other distributions of the Unix operating system evolved at places such as Berkeley (Berkeley Unix or BSD) and MIT. Much of the spread of Unix was a response to the growing commercial control of computer software.

In 1984 Richard Stallman left MIT and began working on the GNU project, a project to create a free version of the Unix operating system. By free, Stallman meant software that

could be freely used, read, modified, and redistributed. Stallman introduced EMACS and GNU, which was short for GNU's Not Unix. His rather radical view of how software should be distributed, led to the GNU Project in 1983, based upon his GNU General Public License, or GPL. His ideas were the foundation of the Free Software Movement in 1983. He articulated the idea of free software, in which free did not mean freedom from cost, but free as in "free speech not free beer". In 1985 interest in the GNU Project had grown and Stallman created the Free Software Foundation (FSF). The FSF successfully built a vast number of useful components, including a C compiler (GCC), an impressive text editor (EMACS), and a host of fundamental tools. However, in the 1990's the FSF was having trouble developing the operating system kernel. Without a kernel their dream of a completely free operating system would not be realized. In 1991 such a kernel was developed by Linus Torvalds, a second year student of Computer Science at the University of Helsinki and a self-taught hacker. For fun, he developed a free operating system and posted the news to the hacker community on August 25, 1991. Linux, as it became known, has undergone many revisions and has grown into a well-known operating system.

The Free Software Definition

Free software is a matter of liberty, not price. To understand the concept, you should think of free as in free speech, not as in free beer.

Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software. More precisely, it refers to four kinds of freedom, for the users of the software:

- * The freedom to run the program, for any purpose (freedom 0).
- * The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.
- * The freedom to redistribute copies so you can help your neighbor (freedom 2).
- * The freedom to improve the program, and release your improvements to the public, so that the whole community benefits (freedom 3). Access to the source code is a precondition for this.

A program is free software if users have all of these freedoms. Thus, you should be free to redistribute copies, either with or without modifications, either gratis or charging a fee for distribution, to anyone anywhere. Being free to do these things means (among other things) that you do not have to ask or pay for permission.

Open Source Resources, ICTCM 2008, San Antonio

<http://www.gnu.org/philosophy/free-sw.html>

In the mid 90's the World Wide Web took hold and there was a growth in the browser industry as the battles between Microsoft and Netscape, an outgrowth of Mosaic, became more competitive. In 1998 the notion of Open Source Software grew out of the free software movement. This was spawned by people like Eric S. Raymond and Bruce Perens as part of the Open Source Initiative (OSI) in 1998. The idea, as espoused by Raymond in his 1997 book, *The Cathedral and the Bazaar*, stated that the source code should be available with each software release, and the code developed between releases should be

restricted to an exclusive group of software developers. GNU Emacs and GCC were examples.

The Open Source Definition

1. Free Redistribution: the software can be freely given away or sold. (This was intended to encourage sharing and use of the software on a legal basis.)
2. Source Code: the source code must either be included or freely obtainable. (Without source code, making changes or modifications can be impossible.)
3. Derived Works: redistribution of modifications must be allowed. (To allow legal sharing and to permit new features or repairs.)
4. Integrity of The Author's Source Code: licenses may require that modifications are redistributed only as patches.
5. No Discrimination Against Persons or Groups: no one can be locked out.
6. No Discrimination Against Fields of Endeavor: commercial users cannot be excluded.
7. Distribution of License: The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. License Must Not Be Specific to a Product: the program cannot be licensed only as part of a larger distribution.
9. License Must Not Restrict Other Software: the license cannot insist that any other software it is distributed with must also be open source.
10. License Must Be Technology-Neutral: no click-wrap licenses or other medium-specific ways of accepting the license must be required.

Open Source Resources, ICTCM 2008, San Antonio
<http://opensource.org/docs/osd>

In the Bazaar model, code is developed over the Internet in public view. Raymond credits Linux creator Linus Torvalds as the inventor of this process. The central thesis is that "given enough eyeballs, all bugs are shallow"; i.e., the more widely available the source code is for public testing, scrutiny, and experimentation, the more rapidly all forms of bugs will be discovered. A recent application of this idea is seen in the popularity of Wikipedia. Wikipedia thrives on using its large community users to contribute and spot errors in the web site entries.

At the same time in 1998 Netscape decided to release the Mozilla source code in support of the Open Source Initiative. The Mosaic Netscape 0.9 browser first appeared on October 13, 1994 as an offshoot of Mosaic by Marc Andreessen, and others from Mosaic. In 1995 Microsoft Internet Explorer appeared in Windows 95 Plus Pack. There was competition between the two browser giants for several years. Then in January 1998 Netscape Communicator 4.0 was provided for free under the Netscape Public License. However, later that year on November 24, 1998 it was announced that Netscape was to be purchased by AOL and on Nov 14, 2000 AOL released Netscape 6.0. Things began to progress fast. On July 15, 2003 Time Warner disbanded Netscape and in Oct 2007 the last release, Netscape Navigator 9.0, came out and on December 28th, AOL announced pending death of Netscape which occurred in March 2008.

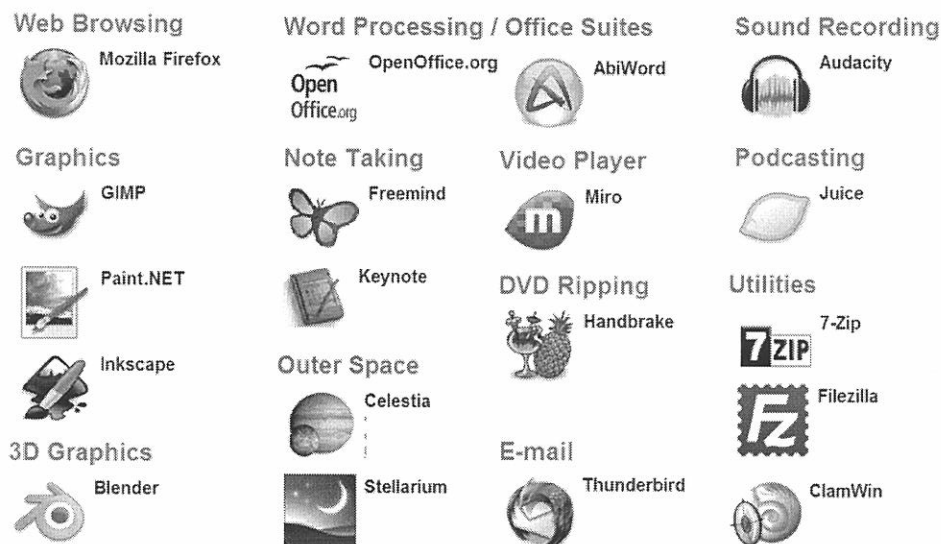
So, what does the demise of Netscape say about open source efforts? In his resignation letter from AOL Jamie Zawinski, a "key figure in opening up Netscape Navigator, and

the person who had come up with the Mozilla name back in 1994 – a combination of the original ‘Mosaic’ and Godzilla”, wrote

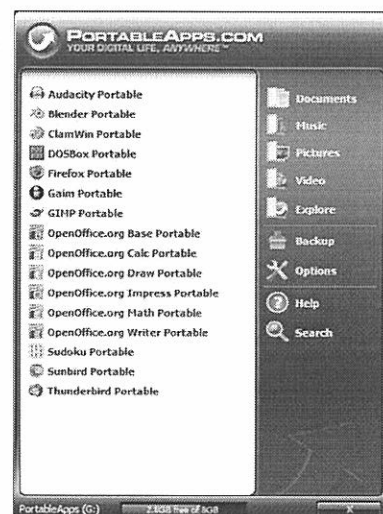
Open source does work, but it is most definitely not a panacea. If there's a cautionary tale here, it is that you can't take a dying project, sprinkle it with the magic pixie dust of “open source,” and have everything magically work out. Software is hard. The issues aren't that simple.”

Common Open Source Software

In recent years the open source movement has grown. There are many decent alternatives to well known proprietary products. The software and its source is located at sites such as sourceforge.net. A few prominent ones are displayed in the figure below.



More recently some open source applications have been trimmed even further so that they can fit on a USB flash drive, or memory key. For example, PortableApps.com (<http://portableapps.com/>) provides a portable suite of applications that you can carry and use on any computer. You can even find trimmed down operating systems, like Linux distributions, which fit on a USB stick, such as the systems offered by <http://www.pendrivelinux.com/>. For Mathematicians, one can install a complete LATEX distribution, Ghostscript, Ghostview software, and install a tex editor on a USB flash memory stick. USBTeX can be found at <http://www.exomatik.net/LaTeX/USBTeXEnglish>



Open Source Resources for Mathematics

The Open Source Initiative has also impacted mathematics software. In Table I we provide a short timeline showing some of the more known mathematics software over the past forty years. In some cases the programs have been replaced by newer versions, or renamed as indicated by the indentation.

Table I. A Timeline of Mathematics Software¹

Reduce 1960s/1968	Maple 1979/1985	LiveMath – 1999
Macsyma 1968/1978	Cayley (1982-1993)	/Expressionist/Theorist
/Maxima 1998	/Magma 1990/1993	/Mathview& MathPlus
(1982 version)	MathCAD 1985	Python 1991,
Axiom 1971/2002	Mathematica 1986/1988	VPython 2000,
MuMath 1970s/1980	PARI/GP 1985	IPython 2005,
/Derive 1988	GAP 1986	SymPy 2006,
/TI-Nspire 2006	MuPad 1989/92	matplotlib, ...
MATLAB late 70's/1984	Fermat 1985/1993	SAGE Feb 24, 2005

Many of these packages have gotten large or expensive. Students cannot afford the costs on top of the increasing costs of textbooks. However, there are other alternatives. Due to space limitations, we list a few open source and free packages which can be used to replace some of the proprietary software. A list of software and links to other sites will be provided at <http://people.uncw.edu/hermanr/OpenSourceMath.htm>.

Computer Algebra Systems

OpenAxiom <http://www.open-axiom.org/index.html>

Maxima <http://maxima.sourceforge.net/>

Yacas <http://yacas.sourceforge.net/homepage.html>

Mathomatic <http://mathomatic.orgserve.de/math/>

Numerical Computing

GNU Octave <http://www.gnu.org/software/octave/>

Scilab <http://www.scilab.org/>

FreeMat <http://freemat.sourceforge.net/>

jMatlab <http://www.jmatlab.org/>

Euler <http://mathsrv.ku-eichstaett.de/MGF/homes/grothmann/euler/index.html>

Geometry

GeoGebra <http://www.geogebra.org/cms/>

Fractint <http://spanky.triumf.ca/www/fractint/fractint.html>land

Xaos <http://wmi.math.u-szeged.hu/xaos/doku.php>

K3DSurf <http://k3dsurf.sourceforge.net/>

¹ http://en.wikipedia.org/wiki/Comparison_of_computer_algebra_systems

Other Applications

Pari/GP <http://pari.math.u-bordeaux.fr/> Number theory

GAP <http://www.gap-system.org/> Discrete math

Macaulay 2 <http://www.math.uiuc.edu/Macaulay2/> algebraic geometry and commutative algebra

Singular <http://www.singular.uni-kl.de/> - commutative algebra, algebraic geometry, and singularity theory.

The R Project <http://www.r-project.org/> - statistical computing and graphics

Python <http://www.python.org/> - a programming environment also useful for computational science with many offshoots capable of interfacing with other programs, providing visualization and data analysis.

There have been several attempts at gathering several open source math programs in one place. Quantian (<http://dirk.eddelbuettel.com/quantian.html>) provides a Linux distribution containing such a collection. In 2005 SAGE (<http://www.sagemath.org/>) was released as a “viable open source alternative to Magma, Maple, Mathematica, and MATLAB. SAGE includes many high-quality open source math packages.” One can even use it online at <https://www.sagenb.org/>

Summary

We have given a brief history and definition of the Open Source Initiative. A list of some of the applications useful for mathematics was provided. There is not sufficient space to cover all of the current offerings and this will likely change in the next few years as more and more people find a need for less expensive and more flexible alternatives to the current packages.

Before closing we should mention the pros and cons of open source. Advantages of open source software are that it is typically less expensive (free), smaller in size, bugs get fixed relatively quickly, and transparency allows quick security updates. Disadvantages of open source may include stability, security, installation may require a computer guru, poor interactivity as graphical interfaces are often the least important feature needed by the developer. However, it can be important in education. Other issues are the poor integration with Microsoft, no customer service, and no warranty.

In spite of the drawbacks, there is something appealing about the freedom that open source software claims to provide. The potential for the wide acceptance of PortableApps.com, SAGE, or Software for Starving Students (<http://softwarefor.org/>) could have a significant impact on how we compute in the next decade.

References

1. Martin A. Goetz, “How ADR Got Into the Software Products Business and Found Itself Competing Against IBM”, 1998.
<http://www.softwarehistory.org/history/Goetz1.html>
2. Dennis M. Ritchie, “The Evolution of the Unix Time-sharing System”, Lecture Notes in Computer Science #79: Language Design and Programming Methodology, Springer-Verlag, 1980.
3. <http://www.gnu.org/gnu/thegnupproject.html>
4. <http://www.opensource.org/history>
5. Moody, Glyn, “The Netscape Story: From Mosaic to Mozilla”, Computer-WorldUK, 12/1/2007,
<http://www.computerworlduk.com/toolbox/opensource/blogs/index.cfm?entryid=310&blogid=14>

Other Web Sites

<http://www.fsf.org/>
<http://www.gnu.org/>
<http://www.opensourcewindows.org/>
<http://www.earlham.edu/~peters/fos/timeline.htm>
http://sage.math.washington.edu/home/wdj/sigsam/opensource_math.html
http://blog.interlinked.org/science/open_source_math_programs.html
<http://people.uncw.edu/hermanr/OpenSourceMath.htm>